UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/967,220	09/28/2001	Dong-Yuan Chen	42390P11197	7151

7590 10/03/2006

Blakely, Sokoloff, Taylor & Zafman
Seventh Floor
12400 Wilshire Boulevard
Los Angeles, CA 90025-1030

EXAMINER	
YIGDALL, MICHAEL J	
ART UNIT	**PAPER NUMBER**
2192	

DATE MAILED: 10/03/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.**	**Applicant(s)**
	09/967,220	CHEN ET AL.
	Examiner	**Art Unit**
	Michael J. Yigdall	2192

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *28 June 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-5,7-15,17-25 and 27-33* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-5,7-15,17-25 and 27-33* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This Office action is responsive to Applicant's submission filed on June 28, 2006. Claims 1-5, 7-15, 17-25 and 27-33 are now pending.

Response to Arguments

2. Applicant's arguments have been considered but are moot in view of the new ground(s) of rejection, as set forth below with reference to Tremblay and Edwards. Applicant's amendment necessitated the new ground(s) of rejection.

Claim Objections

3. Claims 1, 11 and 21 are objected to because the claims recite "detection exception stage" rather than --exception detection stage-- as described in Applicant's specification (see, for example, page 20, lines 5-7). It is also noted that currently amended claim 11 is incorrectly labeled "Previously Presented."

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 1-5, 7, 10-15, 17, 20-25 and 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,134,710 to Levine et al. (art of record, "Levine") in view of U.S. Patent No. 6,542,988 to Tremblay et al. (now made of record, "Tremblay").

With respect to claim 1 (currently amended), Levine discloses a method comprising:

(a) selecting one or more microarchitecture events relating to a microprocessor executing an application process to be monitored by one or more hardware monitors (see, for example, column 2, lines 8-14, which shows selecting events to be recorded by hardware performance monitor counters);

(b) establishing parameters regarding the monitoring of the microarchitecture events by setting one or more monitor control vectors (see, for example, column 2, lines 12-20, which shows setting monitor control registers, i.e. control vectors, to establish monitoring parameters).

Levine further discloses stages of an execution pipeline (see, for example, pipelined instruction cycle 14 in FIG. 4), and discloses that the events are monitored during pipelined execution (see, for example, column 6, lines 4-12), but does not expressly disclose the limitation wherein the one or more monitor control vectors to monitor events in a detection exception stage of an execution pipeline.

However, any stage of the execution pipeline in Levine may comprise exception detection, such as the execute instruction stage (see, for example, FIG. 4). Moreover, Tremblay discloses a pipeline that includes a trap handling stage (see, for example, FIG. 3 and column 9, lines 53-56), which is an exception detection stage (see, for example, column 8, lines 32-33). Events are monitored in this stage (see, for example, column 5, lines 14-31).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to monitor events in an exception detection stage of the execution pipeline in Levine, such as taught by Tremblay.

Levine further discloses:

(c) storing profile data that is captured by the one or more hardware monitors in a first level profile buffer, the first level profile buffer being an architecturally non-visible memory (see, for example, column 10, lines 63-67, which shows storing profile data in the sampled instruction and data address registers, i.e. a first level profile buffer that is an architecturally non-visible memory).

Levine further discloses a load/store unit for reading and writing to memory (see, for example, FIG. 2), but does not expressly disclose the limitation wherein the storing of the profile data being performed in a write-back stage of the execution pipeline.

However, in Levine, writing back to memory is performed in a stage of the execution pipeline, such as in the complete instruction stage (see, for example, FIG. 4). Moreover, Tremblay further discloses that the pipeline includes a write-back stage (see, for example, FIG. 3 and column 9, lines 53-56), in which data from the exception detection stage is stored (see, for example, column 10, lines 45-48).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to store the profile data in a write-back stage of the execution pipeline in Levine, such as taught by Tremblay.

Levine further discloses:

(d) transferring the captured profile data from the first level profile buffer to a second level profile buffer, the second level profile buffer being an architecturally visible storage (see, for example, column 10, line 67 to column 11, line 3, which shows transferring the profile data from the registers to tables in memory, i.e. a second level profile buffer that is an architecturally visible storage);

(e) obtaining the captured profile data from the second level profile buffer (see, for example, column 11, lines 34-53, which shows obtaining the profile data from the tables in memory);

(f) processing the captured profile data (see, for example, column 11, lines 24-34, which shows processing the profile data to generate effective address tables);

(g) identifying a region of interest in the application process for optimization based at least in part on the captured profile data (see, for example, column 12, lines 1-6, which shows analyzing the effective address tables based on the profile data to identify a location or region for optimization); and

(h) optimizing the region of interest in the application process (see, for example, column 13, lines 63 to column 14, line 4, which shows applying the optimizations to the application process).

With respect to claim 2 (original), the rejection of claim 1 is incorporated, and Levine further discloses the limitation wherein setting each monitor control vector comprises setting one or more fields of the monitor control vector to control the monitoring of the microarchitecture event (see, for example, column 2, lines 8-20, which shows setting fields in the control registers to control the event counting or monitoring).

With respect to claim 3 (original), the rejection of claim 2 is incorporated, and Levine further discloses the limitation wherein setting the one or more fields of each monitor control vector includes setting a control field to establish the type of microarchitecture event that is

monitored by a hardware monitor (see, for example, column 8, lines 44-52, which shows setting control fields to select the types of events to be monitored).

With respect to claim 4 (original), the rejection of claim 2 is incorporated, and Levine further discloses the limitation wherein setting the one or more fields of each monitor control vector includes setting a trigger field to control when a microarchitecture event is monitored (see, for example, column 8, lines 23-24 and 35-39, which show setting trigger fields to control when events are counted or monitored).

With respect to claim 5 (currently amended), the rejection of claim 2 is incorporated. Levine discloses setting an interrupt field to cause a handler routine to process the profile data when an event occurs (see, for example, column 8, lines 24-35, which shows the interrupt field in the control register, and column 10, line 67 to column 11, line 5, which shows the handler routine).

Levine does not expressly disclose the limitation wherein setting the one or more fields of each monitor control vector includes storing a pointer in a handler field, the pointer identifying a handler routine to process the captured profile data corresponding to the monitor control vector.

However, a pointer is inherently used in the Levine system to identify the handler routine. The address of the routine must be known in order to invoke the routine and process the captured profile data. It is also well known in the art that a pointer may be stored, for example, in a field of a control vector or register.

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to store the pointer to the handler routine of Levine in a field of the control registers taught by Levine, for the purpose of identifying the address of the routine.

With respect to claim 7 (currently amended), the rejection of claim 1 is incorporated, and Levine further discloses the limitation wherein obtaining the captured profile data for a microarchitecture event from the second level profile buffer occurs when a memory buffer in the second level profile buffer that is assigned for the monitored microarchitecture event is fully allocated (see, for example, column 11, lines 42-53, which shows that the buffer is of a predetermined size and is used in a round-robin fashion, and is therefore fully allocated when the profile data is obtained before being overwritten).

With respect to claim 10 (original), the rejection of claim 1 is incorporated, and Levine further discloses the limitation wherein the microarchitecture event monitored is an instruction cache miss event (see, for example, column 10, lines 2-8, which shows instruction cache miss events, and column 10, lines 15-19 and 34-36, which show counting or monitoring such cache misses).

With respect to claim 11 (currently amended), see the rejection of claim 1 above. The operations recited in claim 11 are analogous to the method steps of claim 1. Note that Levine further discloses a machine-readable medium having stored thereon data representing instructions that, when executed by a processor, cause the processor to perform the recited operations (see, for example, column 1, line 65 to column 2, line 1, and column 2, lines 28-44).

With respect to claim 12 (original), the rejection of claim 11 is incorporated, and the operations recited in the claim are analogous to the method steps of claim 2 (see the rejection of claim 2 above).

With respect to claim 13 (original), the rejection of claim 12 is incorporated, and the operations recited in the claim are analogous to the method steps of claim 3 (see the rejection of claim 3 above).

With respect to claim 14 (original), the rejection of claim 12 is incorporated, and the operations recited in the claim are analogous to the method steps of claim 4 (see the rejection of claim 4 above).

With respect to claim 15 (currently amended), the rejection of claim 12 is incorporated, and the operations recited in the claim are analogous to the method steps of claim 5 (see the rejection of claim 5 above).

With respect to claim 17 (currently amended), the rejection of claim 11 is incorporated, and the operations recited in the claim are analogous to the method steps of claim 7 (see the rejection of claim 7 above).

With respect to claim 20 (original), the rejection of claim 11 is incorporated, and the operations recited in the claim are analogous to the method steps of claim 10 (see the rejection of claim 10 above).

With respect to claim 21 (currently amended), Levine discloses a hardware assisted dynamic optimizer (see, for example, the abstract and FIG. 2), comprising:

(a) an interface to a microprocessor through which the hardware assisted dynamic optimizer establishes parameters regarding the monitoring of one or more microarchitecture events occurring during the execution of an application by the microprocessor (see, for example, column 2, lines 8-20, which shows setting addressable monitor control registers to select events to be recorded and establish monitoring parameters).

Levine further discloses stages of an execution pipeline (see, for example, pipelined instruction cycle 14 in FIG. 4), and discloses that the events are monitored during pipelined execution (see, for example, column 6, lines 4-12), but does not expressly disclose the limitation wherein the monitoring to occur in a detection exception stage of an execution pipeline.

However, any stage of the execution pipeline in Levine may comprise exception detection, such as the execute instruction stage (see, for example, FIG. 4). Moreover, Tremblay discloses a pipeline that includes a trap handling stage (see, for example, FIG. 3 and column 9, lines 53-56), which is an exception detection stage (see, for example, column 8, lines 32-33). Events are monitored in this stage (see, for example, column 5, lines 14-31).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to monitor events in an exception detection stage of the execution pipeline in Levine, such as taught by Tremblay.

Levine further discloses:

(b) one or more handler routines, each handler routine including instructions to process profiles of a monitored microarchitecture event that are captured by the microprocessor (see, for

example, column 10, line 67 to column 11, line 5, which shows a handler routine for processing captured profile data, and column 11, lines 24-34, which shows processing the data to generate effective address tables);

(c) a first level profile buffer to initially store captured profiles, the first level profile buffer being architecturally non-visible (see, for example, column 10, lines 63-67, which shows initially storing profile data in the sampled instruction and data address registers, i.e. a first level profile buffer that is architecturally non-visible).

Levine further discloses a load/store unit for reading and writing to memory (see, for example, FIG. 2), but does not expressly disclose the limitation wherein the storing of the profile data being performed in a write-back stage of the execution pipeline.

However, in Levine, writing back to memory is performed in a stage of the execution pipeline, such as in the complete instruction stage (see, for example, FIG. 4). Moreover, Tremblay further discloses that the pipeline includes a write-back stage (see, for example, FIG. 3 and column 9, lines 53-56), in which data from the exception detection stage is stored (see, for example, column 10, lines 45-48).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to store the profile data in a write-back stage of the execution pipeline in Levine, such as taught by Tremblay.

Levine further discloses:

(d) a second level profile buffer to receive captured profiles from the first level profile buffer, the second level profile buffer being architecturally visible (see, for example, column 10,

line 67 to column 11, line 3, which shows receiving the profile data from the registers in tables in memory, i.e. a second level profile buffer that is architecturally visible); and

(e) one or more optimizers, each optimizer including instructions for optimizing a section of the application, the section of the application being chosen by the hardware assisted dynamic optimizer at least in part based on the captured profiles of a monitored microarchitecture event (see, for example, column 12, lines 1-6, which shows analyzing the effective address tables based on the profile data to identify a location or region for optimization, and column 13, line 63 to column 14, line 4, which shows applying the optimizations to the application).

With respect to claim 22 (original), the rejection of claim 21 is incorporated, and Levine further discloses the limitation wherein each monitor control vector includes a plurality of fields to control the monitoring of the microarchitecture event, the plurality of fields being set by the hardware assisted dynamic optimizer (see, for example, column 2, lines 8-20, which shows setting fields in the control registers to control the event counting or monitoring).

With respect to claim 23 (original), the rejection of claim 22 is incorporated, and Levine further discloses the limitation wherein the plurality of fields includes:

(a) a control field to establish the type of microarchitecture event that is monitored (see, for example, column 8, lines 44-52, which shows setting control fields to select the types of events to be monitored), and

(b) a trigger field to control when the microarchitecture event is monitored (see, for example, column 8, lines 23-24 and 35-39, which show setting trigger fields to control when events are counted or monitored).

Although Levine discloses setting an interrupt field to cause a handler routine to process the profile data when an event occurs (see, for example, column 8, lines 24-35, which shows the interrupt field in the control register, and column 10, line 67 to column 11, line 5, which shows the handler routine), Levine does not expressly disclose the limitation wherein the plurality of fields includes:

(c) a handler field to store a pointer to the handler routine for the microarchitecture event.

However, a pointer is inherently used in the Levine system to identify the handler routine. The address of the routine must be known in order to invoke the routine and process the captured profile data. It is also well known in the art that a pointer may be stored, for example, in a field of a control vector or register.

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to store the pointer to the handler routine of Levine in a field of the control registers taught by Levine, for the purpose of identifying the address of the routine.

With respect to claim 24 (original), the rejection of claim 21 is incorporated, and Levine further discloses the limitation wherein optimizing a section of the application includes increasing the speed of processing of the section of the application (see, for example, column 1, lines 19-23, which shows that delays in executing an application may be caused by long table walks or long cache misses, and see, for example, column 1, lines 58-62, which shows that the optimizations minimize the effects of such table walks and cache misses, i.e. increase the speed of processing the application).

With respect to claim 25 (previously presented), the rejection of claim 21 is incorporated, and Levine further discloses the limitation wherein the hardware assisted dynamic optimizer obtains the captured profiles of the one or more microarchitecture events from the second level profile buffer (see, for example, column 11, lines 34-53, which shows obtaining the profile data from the tables in memory).

With respect to claim 27 (previously presented), the rejection of claim 21 is incorporated, and Levine further discloses the limitation wherein the hardware assisted dynamic optimizer sets conditions for transferring captured profiles from the first level profile buffer to the second level profile buffer (see, for example, column 10, line 67 to column 11, line 3, which shows transferring the profile data from the registers to the tables in memory when an interrupt is serviced, and column 8, lines 24-35, which shows setting conditions for the interrupt).

6. Claims 8, 9, 18, 19 and 28-30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Levine in view of Tremblay, as applied to claims 7, 17 and 27 above, respectively, and further in view of U.S. Patent No. 6,622,300 to Krishnaswamy et al. (art of record, "Krishnaswamy").

With respect to claim 8 (currently amended), the rejection of claim 7 is incorporated, and Levine further discloses setting one or more conditions for transferring captured profile data from the first level profile buffer to the second level profile buffer (see, for example, column 10, line 67 to column 11, line 3, which shows transferring the profile data from the registers to the tables in memory when an interrupt is serviced, and column 8, lines 24-35, which shows setting conditions for the interrupt).

Levine does not expressly disclose setting one or more conditions for obtaining captured profile data when the memory buffer in the second level profile buffer is not fully allocated.

However, Krishnaswamy discloses storing profile data in a buffer and reading the data from the buffer by setting an interrupt condition after a certain number of events, i.e. obtaining the data when the buffer is not fully allocated, so as to sample the profile data nonintrusively without significantly degrading performance (see, for example, column 6, lines 21-45).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to set a condition for obtaining the profile data of Levine when the memory buffer is not fully allocated, so as to sample the profile data nonintrusively without significantly degrading performance, such as taught by Krishnaswamy.

With respect to claim 9 (currently amended), the rejection of claim 8 is incorporated, and Krishnaswamy further discloses receiving an interrupt or special event handler if the memory buffer that is assigned for the microarchitecture event is fully allocated or if a condition for obtaining captured profile data when the memory buffer in the profile buffer is not fully allocated is met (see, for example, column 6, lines 21-45, which shows receiving an interrupt for obtaining the profile data from the buffer when the event-counting condition is met).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use an interrupt for obtaining the profile data of Levine from the memory buffer, so as to sample the traces nonintrusively without significantly degrading performance, such as taught by Krishnaswamy.

With respect to claim 18 (currently amended), the rejection of claim 17 is incorporated, and the operations recited in the claim are analogous to the method steps of claim 8 (see the rejection of claim 8 above).

With respect to claim 19 (currently amended), the rejection of claim 18 is incorporated, and the operations recited in the claim are analogous to the method steps of claim 9 (see the rejection of claim 9 above).

With respect to claim 28 (previously presented), the rejection of claim 27 is incorporated. Levine does not expressly disclose the limitation wherein the hardware assisted dynamic optimizer sets one or more conditions for obtaining captured profiles from the second level profile buffer.

However, Krishnaswamy discloses storing profile data in a buffer and reading the data from the buffer by setting an interrupt condition after a certain number of events, so as to sample the profile data nonintrusively without significantly degrading performance (see, for example, column 6, lines 21-45).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to set a condition for obtaining the profile data of Levine, so as to sample the profile data nonintrusively without significantly degrading performance, such as taught by Krishnaswamy.

With respect to claim 29 (previously presented), the rejection of claim 28 is incorporated, and Levine further discloses the limitation wherein a memory buffer in the second level profile buffer is assigned to a microarchitecture event, and wherein the hardware assisted dynamic

optimizer accesses the profiles of the microarchitecture event when the memory buffer assigned to the microarchitecture event is fully allocated or when a condition for obtaining captured profiles is met (see, for example, column 11, lines 42-53, which shows that the buffer is of a predetermined size and is used in a round-robin fashion, and is therefore fully allocated when the profile data is obtained before being overwritten).

With respect to claim 30 (original), the rejection of claim 29 is incorporated, and Krishnaswamy further discloses the limitation wherein the hardware assisted dynamic optimizer accesses the profiles of a microarchitecture event upon receiving an interrupt or special event handler (see, for example, column 6, lines 21-45, which shows obtaining the profile data upon receiving an interrupt).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to use an interrupt for obtaining the profile data of Levine, so as to sample the profile data nonintrusively without significantly degrading performance, such as taught by Krishnaswamy.

7. Claims 31-33 are rejected under 35 U.S.C. 103(a) as being unpatentable over Levine in view of Tremblay, as applied to claims 1, 11 and 21 above, respectively, and further in view of U.S. Patent No. 6,859,891 to Edwards et al. (now made of record, "Edwards").

With respect to claim 31 (new), the rejection of claim 1 is incorporated. Levine does not expressly disclose the limitation wherein captured profile data is treated as an operand of an instruction for writing back to the first level profile buffer.

However, Edwards discloses capturing profile data that includes operand information in a write-back stage of a pipeline (see, for example, column 12, line 65 to column 13, line 2). The profile data is treated as an operand of an instruction so as to write the profile data over the data bus and thus minimize the need for additional data lines (see, for example, column 7, lines 43-49, and column 13, lines 11-41).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to treat the captured profile data as an operand of an instruction for writing back to the first level profile buffer in Levine, so as to minimize the need for additional data lines, such as taught by Edwards.

With respect to claim 32 (new), the rejection of claim 11 is incorporated, and the limitations recited in the claim are analogous to those of claim 31 (see the rejection of claim 31 above).

With respect to claim 33 (new), the rejection of claim 21 is incorporated, and the limitations recited in the claim are analogous to those of claim 31 (see the rejection of claim 31 above).

Conclusion

8. The prior art made of record and not relied upon is considered pertinent to Applicant's disclosure (see the attached Notice of References Cited).

9. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

10. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Michael J. Yigdall whose telephone number is (571) 272-3707. The examiner can normally be reached on Monday through Friday from 7:30am to 4:00pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

MY

Michael J. Yigdall
Examiner
Art Unit 2192

mjy

TUAN DAM
SUPERVISORY PATENT EXAMINER